Google Analytics
Zopim

France Labs
Open Source Enterprise Search

Navigate...

AJAXFRANCELABS - WIDGETS AND MODULES DOCUMENTATION

SOLR AND CONSTELLIO WIDGETS/MODULES

- Widgets
  - SearchBar
  - Result
  - SearchInformation
  - Table
  - Table Facet Queries
  - Spellcheck
  - AdvancedSearch
  - Menu
  - Slider
  - Suggest
- Modules
  - Pager
  - Autocomplete

CONSTELLIO WIDGETS/MODULES

- Widgets
  - Collection
  - Capsule
- Modules
  - Favorite documents and searches

SEARCHBAR WIDGET

A widget displaying a search bar and search options.

Variables

- **autocomplete:** can take 3 values:
  - **false(default):** no autocomplete.
  - **true:** autocomplete automaticaly created.
  - **AutocomepleteWidget:** an autocomplete widget.
- **autocompleteOptions:** options for the autocomplete widget if autocomplete set to true. Check the code to see the default structure.
- **removeContentButton:** if set to true, creates a button which appears when there is text in the search bar in order to remove it and execute a request (default: false).

Methods

buildWidget: an implementation of the parent class method.
- **beforeRequest:** an implementation of the parent class method.

Note that the Autocomplete widget is also an extension of the Autocomplete plugin from jQuery UI and so it has all its variables and methods too.

RESULT WIDGET

A widget displaying the search result.

Variables

- **pagination:** can take 3 values
  - **false(default):** no pagination.
  - **true:** pagination automaticaly created.
  - **PagerWidget:** a pager widget.

Methods

- **buildWidget:** an implementation of the parent class method.
- **beforeRequest:** an implementation of the parent class method.
- **afterRequest:** an implementation of the parent class method.

SEARCHINFORMATION WIDGET

A widget displaying the search informations like the number of document found, the query execution time, ...

Methods

- **buildWidget:** an implementation of the parent class method.
- **beforeRequest:** an implementation of the parent class method.
- **afterRequest:** an implementation of the parent class method.

TABLE WIDGET

A widget displaying a facet values in a table.

Variables

- **name:** a name for the facet to be displayed.
- **pagination:** can take 3 values:
  - **false(default):** no pagination.
  - **true:** pagination automaticaly created.
  - **PagerWidget:** a pager widget.
- **nbElmToDisplay:** the number of facets to display (default: 10).
- **sort:** can take 3 values
  - **occurences(default):** sort the facets by the number of occurences.

  - **AtoZ:** sort the facets name by alphabetical order.
  - **ZtoA:** sort the facets name in a reversed akphabetical order.
- **maxDisplay:** maximum number of facets (default: 40).
- **checkedOnTop:** if set to true, the checked facets will be sorted on the top before the unchecked facets (default: true).

Methods

- **buildWidget:** an implementation of the parent class method.
- **afterRequest:** an implementation of the parent class method.
- **update:** a method updating the content of the widget.
- **assocTags:** a method maping the search query facets result.
- **sortBy:** the method sorting the facets, takes the sort mode as parameter.
- **clickHandler:** a callback called when a facet is selected or unslected.

TABLE FACET QUERIES WIDGET

A widget displaying facet queries in a table.

Variables

- **name:** a name for the facet to be displayed.
- **field:** the field that will be used in the queries
- **queries:** the list of queries
- **pagination:** can take 3 values:
  - **false(default):** no pagination.
  - **true:** pagination automaticaly created.
  - **PagerWidget:** a pager widget.
- **nbElmToDisplay:** the number of facets to display (default: 10).
- **maxDisplay:** maximum number of facets (default: 40).

SPELLCHECK WIDGET

A widget spellchecking a query string. (eg: can offer you to correct constelli**a** in constelli**o**)
Note: this requires enabling a spellchecker component in the searchhandler of Solr (done in the xml config files), whether you use a pure Solr or a Constellio Solr.

Variables

- **question:** a string displayed before the correction of the search query (defualt: 'Would you try with: ').

Methods

- **buildWidget:** an implementation of the parent class method.
- **beforeRequest:** an implementation of the parent class method.
- **afterRequest:** an implementation of the parent class method.

ADVANCEDSEARCH WIDGET

A widget helping you to build an advanced search form.

The AdvancedSearch widget is built with 2 other classes:

- **AdvancedSearchTable:** which is a part of the advanced search form, containing fields.
- **AdvancedSearchField:** which are the different fields provided by the advanced search.

Variables

- **tables:** an array containing all the tables.
- **resizable:** if set to true, display a resize bar on the bottom of the advanced search form. Requires jQuery UI Resizable plugin.
- **options:** options given to the jQuery UI Resizable plugin.
- **optionsExtend:** if set to true, extends the default parameters defined in this widget (default: true).

Methods

- **buildWidget:** an implementation of the parent class method.
- **beforeRequest:** an implementation of the parent class method.
- **addTable:** add the AdvancedSearchTable object given in parameter to the table array.

AdvancedSearchTable

Variables

- **parent:** the advancedSearch DOM element.
- **elm:** the AdvancedSearchTable DOM element itself.
- **title:** a name for the table.
- **description:** a description for the table.
- **fieldStore:** an array containing the AdvancedSearchField fields related to the table.
- **manager:** a reference to the widget's manager.

Methods

- **init:** an init method.
- **addField:** a method adding an AdvancedSearchField field to the fieldStore array.

AdvancedSearchField

Variables

- **parent:** the advancedSearchTable DOM element.
- **elm:** the AdvancedSearchField DOM element itself.
- **label:** a name for the field.
- **range:** if set to true, will create two inputs instead of one (default: false).
- **description:** a description for the field.
- **multiInput:** if set to true, will add a link which allows the user to add more inputs for this field (default: false).
- **addInputLabel:** text to append to the link adding inputs (default: "Add input").
- **autocomplete:** can take 3 values:
  - **false(default):** no autocomplete.
  - **true:** autocomplete automaticaly created.
  - **AutocomepleteWidget:** an autocomplete widget.
- **autocompleteOptions:** options for the autocomplete widget if autocomplete set to true. Check the code to see the default structure.
- **manager:** a reference to the widget's manager.
- **filter:** if set to true, the widget will add the content of the field to a fq parameter instead of appending it to the q parameter.
- **field:** the field in the search engine index this AdvancedSearchField refers to.

Methods

- **init:** an init method.
- **getValue:** a method returning the value of this field.

MENU WIDGET

A widget building a dropdown menu.

Variables

- **name:** the parent element text to display.
- **menuItems:** a collection of the elements to add to the menu ({name: ..., link: ...}).

Methods

- **buildWidget:** an implementation of the parent class method.

SLIDER WIDGET

A widget building a slider widget, helping you with selecting a values as range or min/max, ...

This widget is an extension of the jQuery UI Slider plugin and also has all its parameters and methods.

Methods

- **buildWidget:** an implementation of the parent class method.

SUGGEST WIDGET

A widget creating a result widget that proposes other suggested documents to your search query. The name may be misleading, this is not the suggest module for an autocomplete, it is rather for a "more like this" behavior.

Variables

- **boots:** a list of boosts to apply. { field1: boost_value1, field2: boost_value2, ... }.
- **resultWidget:** the result widget which will contain the suggestions, internal use.

Methods

- **init:** an implementation of the parent class method.
- **buildWidget:** an implementation of the parent class method.
- **afterRequest:** an implementation of the parent class method.

PAGER MODULE

A module creating a pagination.

Variables

- **source:** the source elements to apply the pagination on. Optional if the elements are retrieved by an Ajax request.
- **prevLabel:** the content of the previous nav button (default: <).
- **nextLabel:** the content of the next nav button (default: >).
- **firstLabel:** the content of the first nav button (default: <<).
- **lastLabel:** the content of the last nav button (default: >>).
- **nbElmToDisplay:** the number of elements to display (default: 10).
- **nbPage:** the number of pages, internal use.
- **nbPageDisplayed:** the number of pages to display in the navigation bar.
- **nbElements:** the number of elements in source, internal use.
- **fastNavigationAlwaysVisible:** if set to true, will always display the navigation button even if they are useless (eg: first and previous button when you are on the first page. Default: false).
- **display:** the css display value of the pagination elements.
- **pageSelected:** the page actually selected (default: 0).

Methods

- **init:** an implementation of the parent class method.
- **updatePages:** the method updating the navigation bar (eg: elements to display)
- **clickHandler:** a callback function called when a click on an element of the navigation is detected.
- **updateList:** a method always called when a click on an element of the navigation is detected. This method is empty by default and must be implemented if you need it.

AUTOCOMPLETE MODULE

This module creates a list of suggestions when you type in an input field. Note: this requires enabling a suggest component as a requesthandler of Solr (done in the xml config files), in case you use a pure Solr. If you are using Constellio, this config is normally already activated in its Solr. For Constellio 1.3, it also requires a modification of the web service servlet in order to reach the Constellio autocomplete service. Contact us if you need help.

This module is also an extension of the jQuery UI Autocomplete plugin and also has all its parameters and methods.

Variables

- **options:** options given to the jQuery UI Autocomplete plugin.
- **optionsExtend:** if set to true, extends the default parameters defined in this module (default: true).
- **render:** an empty variable. You can define it a function changing the way an element is rendering in the autocomplete.
- **field:** if the values retrived must be focused on a specific field of your search engine index. Note that this option will not work with a standard 1.3 Constellio. It requires a modification of its web service. This field variable becomes the variable "f" in the generated query to the server

- **openOnFieldFocus:** if set to true, executes a request to retrieve data when cursor entering the input field (default: false).
- **valueSelectFormat:** a function that format the data selected in the autocomplete results (eg: if you want to put it between double quotes).
- **singleValue:** if set to true, when a value is choson in the autocomplete result, the actual value in the input field will be replaced by the selected one.
- **searchForItSelf:** if set to true, also search the term itself (default: false). This means that the list will contain the word currently typed in addition to the words which have at least one more letter

Methods

- **init:** an implementation of the parent class method.

COLLECTION WIDGET

A widget displaying your constellio's collections. Obviously, this widget works only for Constellio. Note that this requires that your manager was initialised with the array of Constellio collections. See AbstractManager.collections for that. Note also that it requires a modification of the Constellio 1.3 web service servlet. Contact us to know what to modify.

Methods

- **buildWidget:** an implementation of the parent class method.

CAPSULE WIDGET

A widget that gives you tips about your search. Does not work for Solr. Note that this only works for Constellio 1.3, with a modified web service in order to support the request.

Methods

- **init:** an implementation of the parent class method.
- **buildWidget:** an implementation of the parent class method.
- **beforeRequest:** an implementation of the parent class method.
- **afterRequest:** an implementation of the parent class method.

FAVORITE DOCUMENTS AND SEARCHES

A module adding the possibility to create a favorite system on your searches and documents. Does not work for Solr. Note that this only works for Constellio 1.3, with a modified web service in order to support the request.

Variables

- **userId:** the id of the user manipulating its favorites. This is actually optional if your server is managing the user sessions.

Methods

- **beforeRequest:** an implementation of the parent class method.
- **afterRequest:** an implementation of the parent class method.
- **removeFavoriteDocument:** takes as parameter the id of the favorite document and a callback method.
- **removeFavoriteSearch:** takes as parameter the id of the favorite search and a callback method.
- **removeSearchInHistory:** takes as parameter the id of the search to remove in the history and a callback method.
- **getFavoriteDocuments:** takes as parameter a callback function, the first element to start on and the number of elements to retrieve.
- **getFavoriteSearches:** takes as parameter a callback function, the first element to start on and the number of elements to retrieve.
- **getSearchHistory:** takes as parameter a callback function, the first element to start on and the number of elements to retrieve.

- Introduction
- Documentation
- Widgets and modules
- Tutorial
- Downloads
- Licence

Copyright © 2014 France Labs / Template HTML Little NEKO